Luminosity DB next (2) year(s)

Marco Verzocchi
Fermilab

Data volume

Expect it to grow linearly with time (but slope reduced since plenty of useless messages/flags removed, slope should be taken from last few weeks)

I am assuming that we are NOT going to merge the VME/NIM columns into a single one

Expect usage from users to grow once we make the DB calculation of the luminosity the default (soon)

Reprocessing/replacing old data

I am slowly reprocessing the old data with the goal of finding missing/wrong data in the database and add/replace the information when necessary

Initial discussions (with Amber, a long time ago) indicated that easiest thing would be to wipe the DB and fill it again from scratch

I don't believe this is necessary, fraction of data which needs to be replaced is small (few % at most), can be done with normal update procedures

This is not going to be finished before the end of the summer, will coordinate with DBA (do we want to stop backup when/if I replace few % of the data ?) when ready

The real issues are the DBservers

Running two versions of DB servers

"old": serves access from farms (Python + C++ wrapper) or users (Python only) to instantaneous luminosity information

"new": for luminosity calculation

In principle would like to use only 1 code version ("new")

Switch to "new" delayed for 2 reasons:

- **stability of farm production**
- **"new" not ready/fully certified**

C++ vs Python servers: current configuration is probably fine (farms rely heavily on caching, the rest of the operations probably not)

Only 1 version of DBservers

Offline production code (farms) does not need to be recompiled to deal with new DBserver (fortunate consequence of some sloppy coding....)

Still I would prefer to have a natural break in processing to do the transition (to avoid being blamed for bringing down the farm for a couple of days while we redeploy software)

Will need one more test, probably best time is a 1 week shutdown of Tevatron foreseen for this summer

People involved: Marco + Steve (or Robert Illingworth)

Usage of DB servers

Have had at least 2 (series of) incidents during the past year in which heavy access to instantaneous luminosity information brought down the lumi DB servers (reason for decoupling this type of access from the luminosity calculation)

Access from D0 farms is optimized for caching information and has never been a problem

We overlay data events (zero bias events) on our MC samples and therefore need luminosity access also for MC. This was a major problem, which was solved by reorganizing the processing of zero bias events to optimize access to cache

Old MC samples are a problem, they need to access the DB for each event (in a very inefficient way)

Only solution is limit on number of jobs processing old MC

Usage of DB servers

Unfortunately some users not aware of this, they bring the DB servers down, Steve complains to Marco, Marco reminds the appropriate people of the problem (turn around this last time was quite fast)

Unfortunately this has also happened for another user code (last time few weeks ago), where access to the luminosity database was not needed at all

Analyzed log files from DB servers and found out all people accessing the instantaneous luminosity information from the DB servers, contacted each of them
In last few days there is only 1 user accessing the instantaneous luminosity information, I need to check what this user is doing (may be part of legitimate reprocessing of old MC samples)

Unfortunately the reprocessing of old MC samples is not going away

Luminosity calculation

Will probably make the DB based luminosity calculation the default in the very near future

Separate DB server so no interference on farm production

However two worries

- **load on Oracle DB**
- **load on DB server**

Need to make 1 final test, what is the maximum number of luminosity calculations which can be performed simultaneously

I would prefer installing a queuing mechanism to limit the number of luminosity calculations which are performed simultaneously (1 or 2). Would use a batch queue on clued0 for this

Need to gain experience